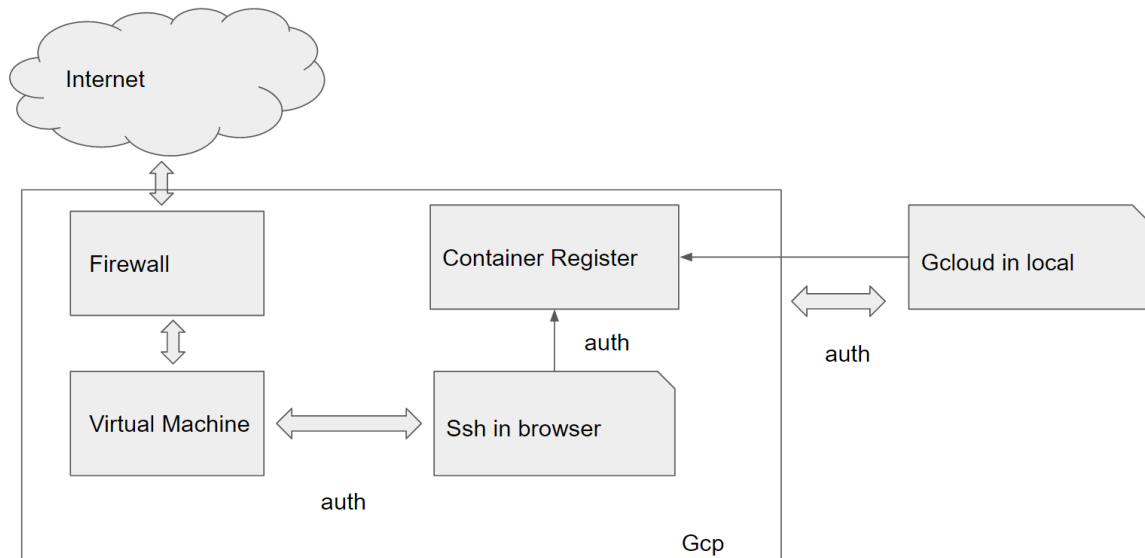


Modelo en gcp ,utilizado

Gcp deploy



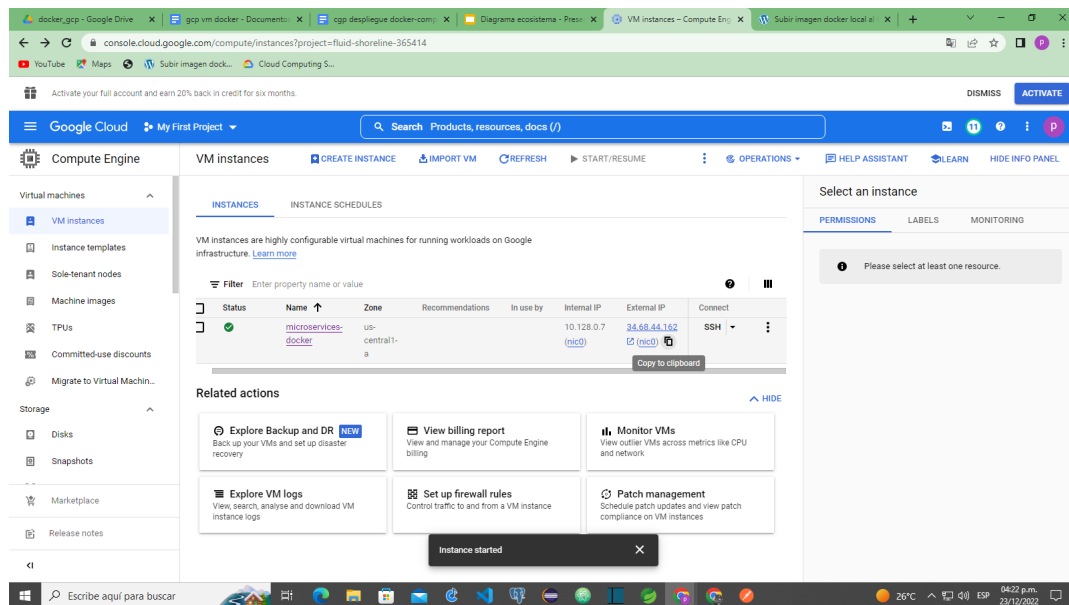
Virtual machine:

Se debe tener una cuenta en gcp , y crear un proyecto

Se debe crear una máquina virtual con similares características que el local, para poder correr todos los contenedores.

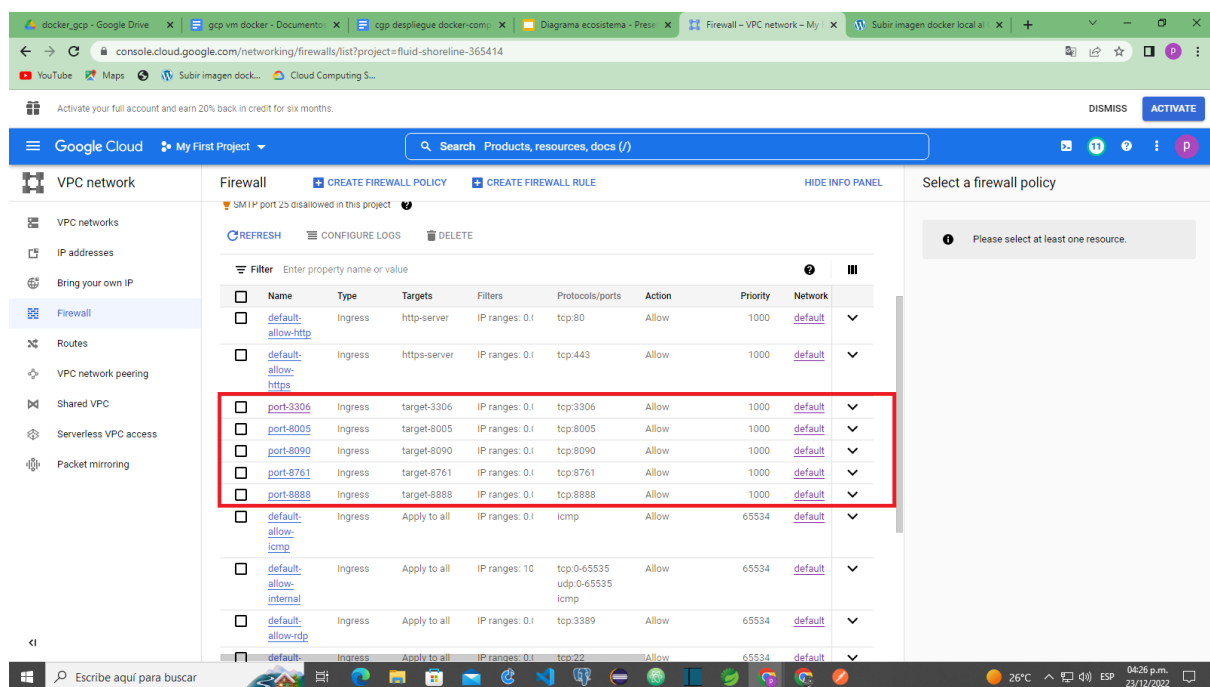
Se debe configurar el use del sistema operativo optimizado para contenedores

Cada vez que se inicia la VM, se publica la IP externa para acceder a la Vm.



Firewall

Se debe configurar para poder acceder a l Vm desde la internet (ver anexo 1)



para el uso del ecosistema sólo se debe configurar el acceso a zuul, port 8090
los demás puertos se usaron para la puesta en marcha del ecosistema

Ssh in browser

Se utiliza para conectarse a la vm , el os es un ubuntu optimizado para contenedores
solo tiene el docker instalado
se debe instalar el docker-compose con los siguiente comandos

```
curl -s
https://gist.githubusercontent.com/kurokobo/25e41503eb060fee8d8bec1dd859eff3/raw/0d7cd29472
f0eaa26ce424071456ad84b24fb318/installer.sh | bash
```

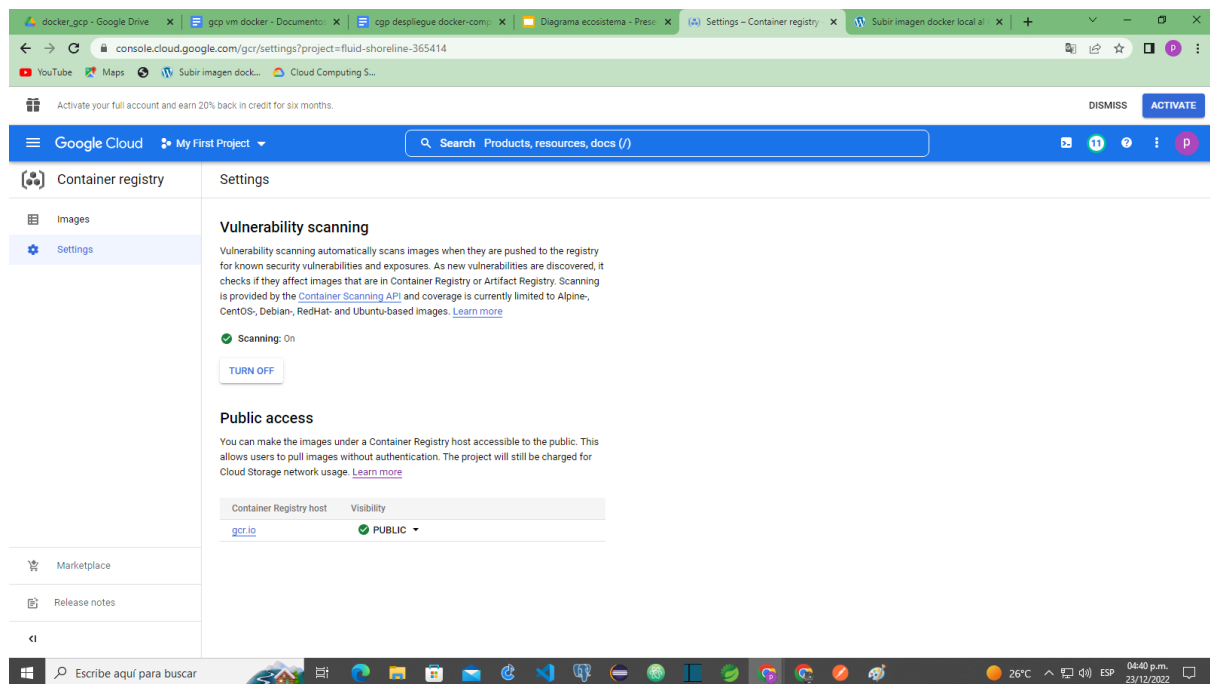
```
source ~/.bashrc
```

para verificar la correcta instalación : docker-compose version

Container registry

Aca se deben subir las imágenes desde el local, para luego levantar los contenedores desde la Vm

Para evitar la autenticación al acceder al registro se configuró en público



gcloud in local

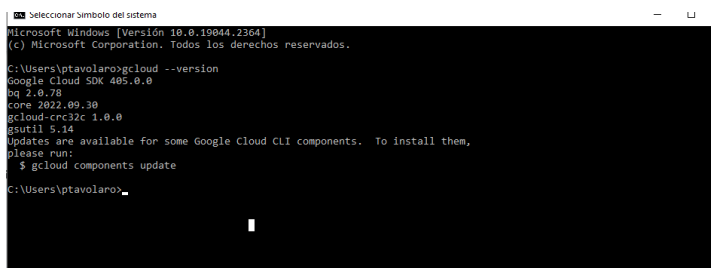
Instalar en el local gcloud cli

Este se utiliza para subir la imagenes del local al container register .

Subir imagenes al container registry (ver anexo 3)

para verificar si esta instalado gcloud cli

gcloud --version



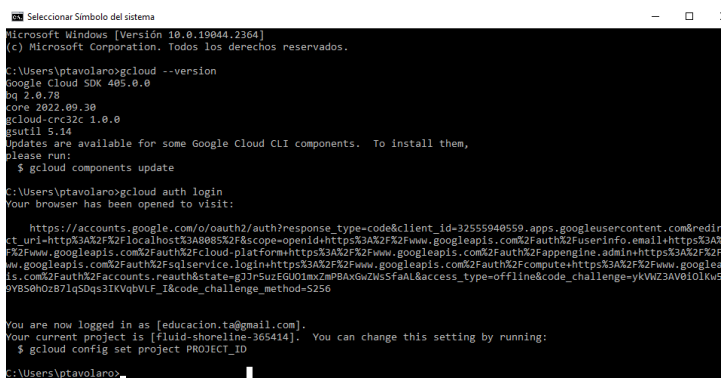
```
Microsoft Windows [Versión 10.0.19044.2364]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\ptavolaro>gcloud --version
Google Cloud SDK 405.0.0
bq 2.0.78
core 2022.09.30
gcloud-crc32c 1.0.0
gsutil 5.14
Updates are available for some Google Cloud CLI components. To install them,
please run:
  $ gcloud components update

C:\Users\ptavolaro>
```

pedir autorización para conectarse a gcp

gcloud auth login



```
Microsoft Windows [Versión 10.0.19044.2364]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\ptavolaro>gcloud --version
Google Cloud SDK 405.0.0
bq 2.0.78
core 2022.09.30
gcloud-crc32c 1.0.0
gsutil 5.14
Updates are available for some Google Cloud CLI components. To install them,
please run:
  $ gcloud components update

C:\Users\ptavolaro>gcloud auth login
Your browser has been opened to visit:
  https://accounts.google.com/o/oauth2/auth?response_type=code&client_id=32555940559.apps.googleusercontent.com&redire
ct_uri=http%3A%2F%2Flocalhost%3A8085%2F&scope=openid+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email+https%3A%2
F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fw
ww.googleapis.com%2Fauth%2Fsqlservice.login+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleap
is.com%2Fauth%2Faccounts.reauth&state=g3Jle5uzEQ0ImxZmP8AXGwZw5FaAL&access_type=offline&code_challenge=yKVMZ3AVB10Ikw59
9VB50H0z871q5Dq531KVbVLF_I&code_challenge_method=S256

You are now logged in as [educacion.ta@gmail.com].
Your current project is [fluid-shoreline-365414]. You can change this setting by running:
  $ gcloud config set project PROJECT_ID

C:\Users\ptavolaro>
```

El proyecto está corriendo en el local con docker-compose el archivo que se usó para levantar se puede ver en el anexo 2

para ver el nombre del proyecto activo
gcloud config get project

fluid-shoreline-365414

para activar el proyecto si no esta activo

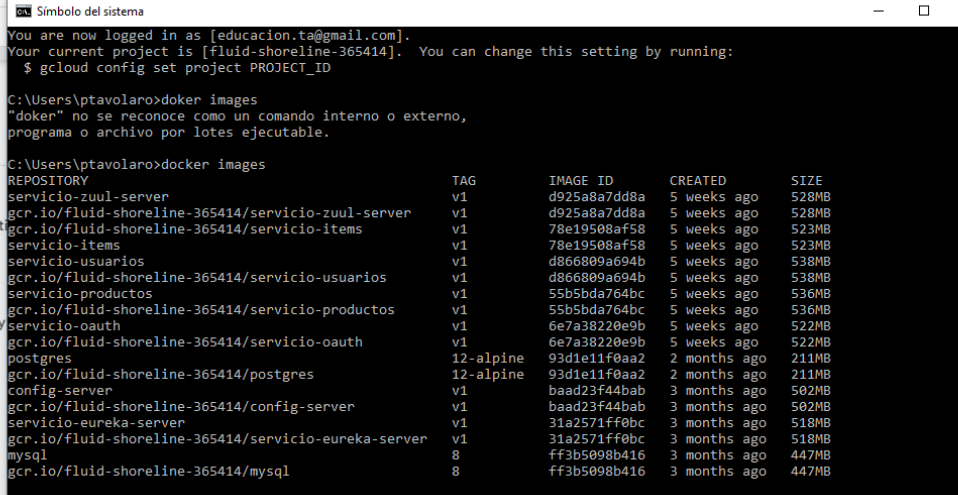
gcloud config set project fluid-shoreline-365414

para poder subir la imagenes al gcr se debe poner la ruta donde se subir
la ruta es gcr.io/[nombre_proyecto]/[nombre_imagen]

para esto por cada imagen local se debe crear un tag con la ruta correspondiente

```
docker tag config-server:v1 gcr.io/fluid-shoreline-365414/config-server:v1
docker tag servicio-eureka-server:v1 gcr.io/fluid-shoreline-365414/servicio-eureka-server:v1
docker tag servicio-productos:v1 gcr.io/fluid-shoreline-365414/servicio-productos:v1
docker tag servicio-items:v1 gcr.io/fluid-shoreline-365414/servicio-items:v1
docker tag servicio-usuarios:v1 gcr.io/fluid-shoreline-365414/servicio-usuarios:v1
docker tag servicio-oauth:v1 gcr.io/fluid-shoreline-365414/servicio-oauth:v1
docker tag servicio-zuul-server:v1 gcr.io/fluid-shoreline-365414/servicio-zuul-server:v1
```

al listar la imágenes se verá



```
Símbolo del sistema
You are now logged in as [educacion.ta@gmail.com].
Your current project is [fluid-shoreline-365414]. You can change this setting by running:
$ gcloud config set project PROJECT_ID

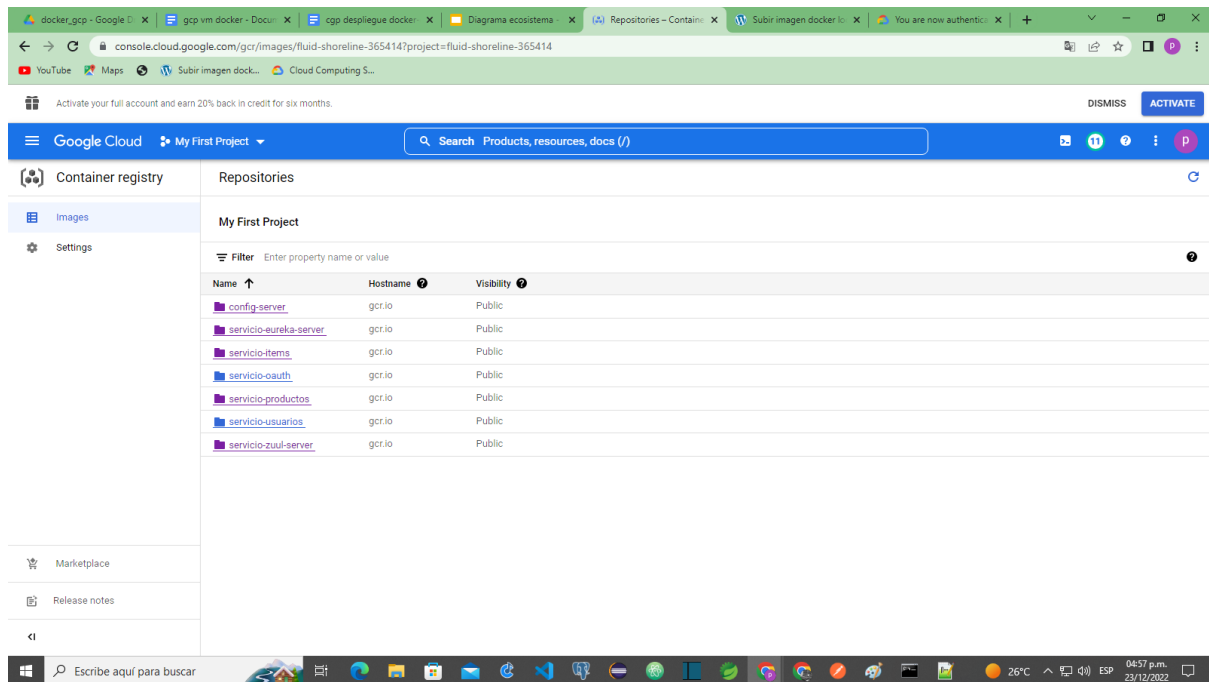
C:\Users\ptavolaro>docker images
REPOSITORY                                TAG                IMAGE ID           CREATED           SIZE
servicio-zuul-server                      v1                d925a8a7dd8a      5 weeks ago      528MB
gcr.io/fluid-shoreline-365414/servicio-zuul-server  v1                d925a8a7dd8a      5 weeks ago      528MB
gcr.io/fluid-shoreline-365414/servicio-items       v1                78e19508af58      5 weeks ago      523MB
servicio-items                           v1                78e19508af58      5 weeks ago      523MB
servicio-usuarios                         v1                d866809a694b      5 weeks ago      538MB
gcr.io/fluid-shoreline-365414/servicio-usuarios   v1                d866809a694b      5 weeks ago      538MB
servicio-productos                       v1                55b5bda764bc      5 weeks ago      536MB
gcr.io/fluid-shoreline-365414/servicio-productos  v1                55b5bda764bc      5 weeks ago      536MB
servicio-oauth                           v1                6e7a38220e9b      5 weeks ago      522MB
gcr.io/fluid-shoreline-365414/servicio-oauth      v1                6e7a38220e9b      5 weeks ago      522MB
postgres                                 12-alpine         93d1e11f0aa2      2 months ago     211MB
gcr.io/fluid-shoreline-365414/postgres            12-alpine         93d1e11f0aa2      2 months ago     211MB
config-server                            v1                baad23f44bab      3 months ago     502MB
gcr.io/fluid-shoreline-365414/config-server       v1                baad23f44bab      3 months ago     502MB
servicio-eureka-server                   v1                31a2571ff0bc      3 months ago     518MB
gcr.io/fluid-shoreline-365414/servicio-eureka-server v1                31a2571ff0bc      3 months ago     518MB
mysql                                    8                 ff3b5098b416      3 months ago     447MB
gcr.io/fluid-shoreline-365414/mysql              8                 ff3b5098b416      3 months ago     447MB
```

Nota: las imágenes de las bases de datos se pueden subir directo desde la VM , en la consola se ven con su nuevo tag, pero no es necesario.

para subir la imagenes al container registry

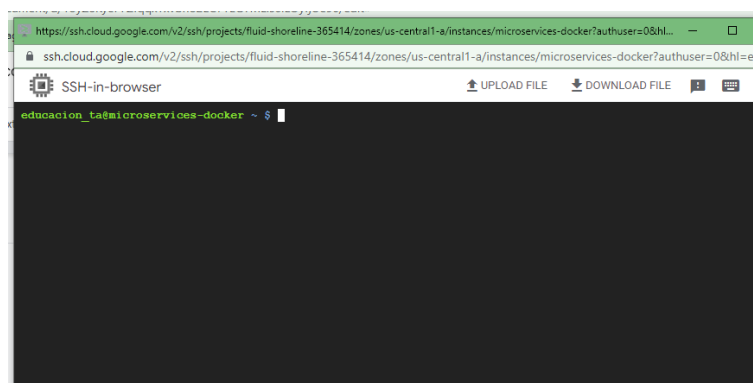
```
docker push gcr.io/fluid-shoreline-365414/config-server:v1
docker push gcr.io/fluid-shoreline-365414/servicio-eureka-server:v1
docker push gcr.io/fluid-shoreline-365414/servicio-productos:v1
docker push gcr.io/fluid-shoreline-365414/servicio-items:v1
docker push gcr.io/fluid-shoreline-365414/servicio-usuarios:v1
docker push gcr.io/fluid-shoreline-365414/servicio-oauth:v1
docker push gcr.io/fluid-shoreline-365414/servicio-zuul-server:v1
```

si vamos a la consola de gcp



Subir imagenes a la VM

conectar por ssh a la vm



corremos los contenedores de cada imagenes desde docker

```
docker run -p 8888:8888 --network springcloud -d config-server:v1
```

```
docker run -p 8761:8761 --network springcloud -d servicio-eureka-server:v1
```

```
docker run -p 3306:3306 --network springcloud -e  
MYSQL_ROOT_PASSWORD=sasa -e MYSQL_DATABASE=db_springboot_cloud  
-d microservicios-mysql8:8
```

```
docker run -p 8761:8761 --network springcloud -d servicio-items:v1
```

```
docker pull postgres:12-alpine
```

```
docker run -p 5432:5432 --name microservicios-postgres12 --network springcloud -e  
POSTGRES_PASSWORD=sasa -e POSTGRES_DB=db_springboot_cloud -d  
postgres:12-alpine
```

```
docker run -P --network springcloud servicio-usuarios:v1
```

```
docker run -P --network springcloud  
gcr.io/fluid-shoreline-365414/servicio-usuarios:v1
```

```
docker run -p 9100:9100 --network springcloud  
gcr.io/fluid-shoreline-365414/servicio-oauth:v1
```

```
docker run -p 8090:8090 --network springcloud  
gcr.io/fluid-shoreline-365414/servicio-zuul-server:v1
```

una vez corriendo los contenedores

para todos con

```
docker stop [Id_container ]
```

cómo se subieron con la ruta a el container registry como nombre de imagen se debe renombrar , para que queden los nombres que están configurados en los jar de los microservicios

```
docker tag gcr.io/fluid-shoreline-365414/servicio-usuarios:v1 servicio-usuarios:v1
```

```
docker tag gcr.io/fluid-shoreline-365414/servicio-oauth:v1 servicio-oauth:v1
```

```
docker tag gcr.io/fluid-shoreline-365414/servicio-zuul-server:v1
```

```
servicio-zuul-server:v1
```

aca solo hay tres pero se debe hacer con todos

conviene borrar los nombres con las rutas

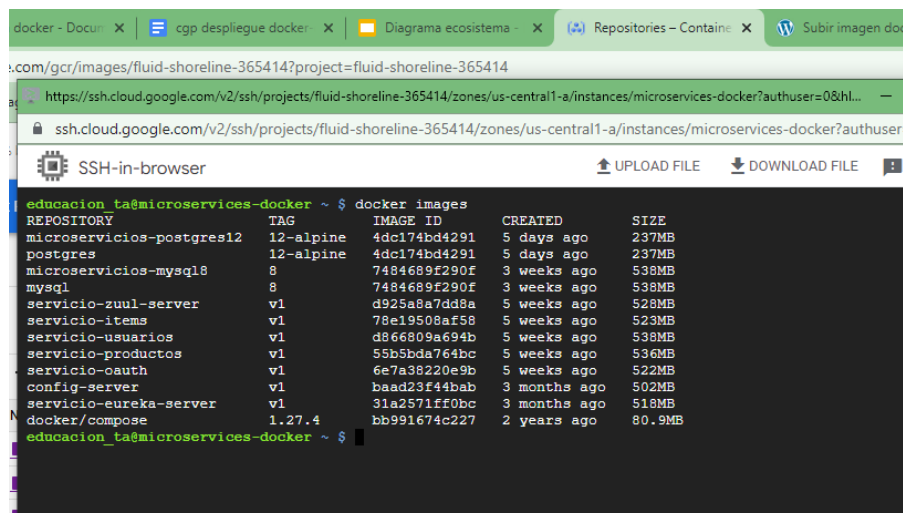
```
docker rmi gcr.io/fluid-shoreline-365414/servicio-usuarios:v1
```

```
docker rmi gcr.io/fluid-shoreline-365414/servicio-oauth:v1
```

aca solo hay dos pero se debe hacer con todos

las imágenes deben tener los nombres que tiene en el docker-compose.yml(anexo 2)

docker images



```
educacion_ta@microservices-docker ~ $ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
microservicios-postgres12  12-alpine  4dc174bd4291  5 days ago   237MB
postgres            12-alpine  4dc174bd4291  5 days ago   237MB
microservicios-mysql8    8          7484689f290f  3 weeks ago   538MB
mysql               8          7484689f290f  3 weeks ago   538MB
servicio-zuul-server    v1         d925a8a7dd8a  5 weeks ago   528MB
servicio-items         v1         78e19508af58  5 weeks ago   523MB
servicio-usuarios      v1         d866809a694b  5 weeks ago   538MB
servicio-productos     v1         55b5bda764bc  5 weeks ago   536MB
servicio-oauth         v1         6e7a38220e9b  5 weeks ago   522MB
config-server         v1         baad23f44bab  3 months ago  502MB
servicio-eureka-server v1         31a2571ff0bc  3 months ago  518MB
docker/compose        1.27.4     bb991674c227  2 years ago   80.9MB
```

aca se ven las dos bases de datos con el nombre original , esto es porque esta imagen fue capturada luego de levantar con docker-compose , y este levanta las imágenes originales

Levantando con docker-compose

crear docker-compose.yml

se debe crear el archivo docker-compose.yml en la Vm

desde la Ssh

```
touch docker-compose.yml
```

```
vim docker-compose.yml
```

pegar código del anexo 2

esc :w :q (para salir y guardar)

Algunos microservicios tardan en levantarse, para seguridad conviene ir levantando de a uno hasta conocer el comportamiento de los mismos

```
docker-compose up -d [nombre_imagen] -para no ver la consola de java
```

```
docker-compose up [nombre_imagen] -para ver la consola de java
```

para parar todos los microservicios

```
docker-compose stop
```

anexo 1-Configurar el firewall para la vm

<https://www.youtube.com/watch?v=sPbBVim9Zbk&t=86s>

anexo 2-docker-compose.yml

```
version: '3.7'
services:
  config-server:
    image: config-server:v1
    ports:
      - "8888:8888"
    restart: always
    networks:
      - springcloud
  servicio-eureka-server:
    image: servicio-eureka-server:v1
    ports:
      - "8761:8761"
    restart: always
    networks:
      - springcloud
  microservicios-mysql8:
    image: mysql:8
    ports:
      - "3306:3306"
    restart: always
    networks:
      - springcloud
    environment:
      MYSQL_DATABASE: db_springboot_cloud
      MYSQL_ROOT_PASSWORD: sasa
  microservicios-postgres12:
    image: postgres:12-alpine
    ports:
      - "5432:5432"
    restart: always
    networks:
      - springcloud
    environment:
      POSTGRES_DB: db_springboot_cloud
      POSTGRES_PASSWORD: sasa
  servicio-productos:
    image: servicio-productos:v1
    restart: always
    networks:
      - springcloud
    depends_on:
      - config-server
      - servicio-eureka-server
      - microservicios-mysql8
  servicio-items:
    image: servicio-items:v1
    ports:
      - "8002:8002"
      - "8005:8005"
      - "8007:8007"
    restart: always
    networks:
      - springcloud
    depends_on:
```

- config-server
- servicio-eureka-server
- servicio-productos

servicio-usuarios:

image: servicio-usuarios:v1

restart: always

networks:

- springcloud

depends_on:

- config-server
- servicio-eureka-server
- microservicios-postgres12

servicio-oauth:

image: servicio-oauth:v1

ports:

- "9100:9100"

restart: always

networks:

- springcloud

depends_on:

- config-server
- servicio-eureka-server
- servicio-usuarios

servicio-zuul-server:

image: servicio-zuul-server:v1

ports:

- "8090:8090"

restart: always

networks:

- springcloud

depends_on:

- config-server
- servicio-eureka-server
- servicio-productos
- servicio-items
- servicio-usuarios
- servicio-oauth

networks:

springcloud:

anexo 3- Subir imagen al container registry

imagen al container registry

<http://www.notasmentalesdeunsysadmin.com/2020/05/subir-imagen-docker-local-al-container-registry-de-gcp/>

Google Cloud Platform|GCP|GCR| Build And Push Docker Image to Google Cloud Container Registry

<https://www.youtube.com/watch?v=n5jqM08VTI4>

anexo 4 - comando útiles

listar proyectos en gcp

gcloud projects list

habilitar container register

gcloud services enable containerregistry.googleapis.com \

gcloud auth configure-docker

listar las redes

docker network ls

para container

docker stop my_container

remove images

docker rmi ff3b5098b416

===== config-server

.\mvnw clean package

docker build -t config-server:v1 .

docker network create spring-microservicios

docker run -p 8888:8888 --name config-server --network springcloud

config-server:v1

===== servicio-eureka-server

.\mvnw clean package

docker build -t servicio-eureka-server:v1 .

docker run -p 8761:8761 --name servicio-eureka-server --network springcloud

servicio-eureka-server:v1

===== mysql

docker pull mysql:8

```
docker run -p 3306:3306 --name microservicios-mysql8 --network springcloud
-e MYSQL_ROOT_PASSWORD=sasa -e
MYSQL_DATABASE=db_springboot_cloud -d mysql:8
docker logs -f microservicios-mysql8
```

===== postgresql

```
docker pull postgres:12-alpine
docker run -p 5432:5432 --name microservicios-postgres12 --network
springcloud -e POSTGRES_PASSWORD=sasa -e
POSTGRES_DB=db_springboot_cloud -d postgres:12-alpine
docker logs -f microservicios-postgres12
```

===== springboot-servicio-productos

```
.\mvnw clean package -DskipTests
```

```
docker build -t servicio-productos:v1 .
docker run -P --network springcloud servicio-productos:v1
```

===== springboot-servicio-zuul-server

```
.\mvnw clean package -DskipTests
```

```
docker build -t servicio-zuul-server:v1 .
docker run -p 8090:8090 --network springcloud servicio-zuul-server:v1
```

===== springboot-servicio-usuarios

```
.\mvnw clean package -DskipTests
```

```
docker build -t servicio-usuarios:v1 .
docker run -P --network springcloud servicio-usuarios:v1
```

===== springboot-servicio-oauth

```
.\mvnw clean package -DskipTests
```

```
docker build -t servicio-oauth:v1 .
docker run -p 9100:9100 --network springcloud servicio-oauth:v1
```

===== springboot-servicio-item

```
.\mvnw clean package -DskipTests
```

```
docker build -t servicio-items:v1 .
docker run -p 8002:8002 -p 8005:8005 -p 8007:8007 --network springcloud
servicio-items:v1
```

===== rabbitmq

```
docker pull rabbitmq:3.8-management-alpine
docker run -p 15672:15672 -p 5672:5672 --name microservicios-rabbitmq38
--network springcloud -d rabbitmq:3.8-management-alpine
```

```
docker logs -f microservicios-rabbitmq38
```

===== zipkin

```
docker build -t zipkin-server:v1 .
docker run -p 9411:9411 --name zipkin-server --network springcloud -e
RABBIT_ADDRESSES=microservicios-rabbitmq38:5672 -e
STORAGE_TYPE=mysql -e MYSQL_USER=zipkin -e MYSQL_PASS=zipkin
-e MYSQL_HOST=microservicios-mysql8 zipkin-server:v1
docker logs -f zipkin-server
```

===== Otros comandos

detener y eliminar todos los contenedores:

```
docker stop $(docker ps -q)
docker rm $(docker ps -a -q)
```